

OPTIMASI FUNGSI KEANGGOTAAN FUZZY BERBASIS ALGORITMA MODIFIED PARTICLE SWARM OPTIMIZATION

Rimbun Siringoringo¹, Zakarias Situmorang²

¹ringorbnsrg@gmail.com, ²zakarias65@yahoo.com

¹ Mahasiswa Magister Teknik Informatika, Univesitas Sumatera Utara

² Dosen Fakultas Ilmu Komputer, Universitas Katolik Santo Thomas Medan

ABSTRAK

Pada penelitian ini optimasi berbasis algoritma *Modified Particle Swarm Optimization* (MPSO) diterapkan untuk mengoptimasi fungsi keanggotaan fuzzy. Terdapat dua metode MPSO yang diterapkan yaitu metode *Linear Decreasing Inertia Weight* (LDIW) dan *Constriction Factor Method* (CFM). Masing-masing metode tersebut diuji dengan 10 kali percobaan pada dua jenis jumlah *particle* yaitu 50 dan 20 *particle*. Dari hasil pengujian diperoleh bahwa pada jumlah *particle* yang sama, CFM memperoleh nilai *global best fitness* yang lebih optimal daripada metode LDIW. Pengujian sebanyak 10 kali percobaan dan menerapkan 50 *particle*, pada percobaan pertama diperoleh nilai *global best fitness* yaitu 1,4; 1,4; 2,36 dan 3,28 untuk masing-masing variabel produktifitas, keterisolasian, hubungan sosial dan aksesibilitas. Pengujian sebanyak 10 kali percobaan dan menerapkan 20 *particle* diperoleh nilai *global best fitness* yaitu 2,34; 2,40; 2,37 dan 3,36 untuk masing-masing variabel. Di sisi lain metode CFM memperoleh hasil konvergensi yang lebih cepat dari pada metode LDIW. Pengujian pada 100 *swarm* metode LDIW menemukan *global best fitness* pada *swarm* 91, 84, 54 dan 38 untuk masing-masing variabel, sementara dengan metode CFM menemukan *global best fitness* pada *swarm* 81, 23, 34 dan 23.

Kata Kunci : Logika Fuzzy, *Modified Particle Swarm Optimization*, Fungsi Keanggotaan

1. PENDAHULUAN

Fungsi keanggotaan memegang peranan yang sangat krusial pada sistem pengambilan keputusan berbasis logika fuzzy. Fungsi keanggotaan merupakan inti dari disain pengambilan keputusan dengan logika fuzzy [1]. Hal senada juga dikemukakan oleh [2] bahwa pemilihan fungsi keanggotaan merupakan kunci utama pada logika fuzzy karena disain fungsi keanggotaan sangat mempengaruhi konsistensi, ketepatan dan kualitas keputusan yang diselesaikan pada sistem tersebut.

Optimasi fungsi keanggotaan fuzzy merupakan bidang kajian riset yang menarik minat para peneliti dalam rangka mencari alternatif perbaikan fungsi

keanggotaan fuzzy. Terdapat berbagai penelitian terkait yang menerapkan teknik optimasi pada perbaikan fungsi keanggotaan fuzzy. Penelitian tersebut adalah *Simulated Annealing* (SA) [3], *Clonal Selection Algorithm* (CSA) [4], Jaringan Saraf Tiruan (JST) [5] dan *Artificial Ant Colony optimization* (AACO) [6].

Particle Swarm Optimization (PSO) adalah metode yang sering diterapkan untuk memecahkan masalah optimasi pada kasus *nonlinier*. PSO menghasilkan bobot yang lebih optimal sebagai alternatif *Back Propagation* (BP) dalam pelatihan JST [7]. PSO memberikan hasil yang optimal pada *clusters centre* sebagai alternatif K-Means pada masalah klasifikasi pola atau *pattern classification* [8]. Kombinasi PSO dengan *Support Vector Machine* (SVM) pada optimasi penjadwalan proyek diperoleh performa dan hasil yang lebih optimal [9]. Kombinasi PSO dengan *Statistical Clustering* (SC) pada masalah klasifikasi fitur menunjukkan bahwa kedua algoritma dapat memilih fitur dengan jumlah yang jauh lebih kecil dan mencapai kinerja klasifikasi yang lebih baik [10].

Pada metode PSO klasik, nilai bobot inersia atau *inertia weight* dibuat konstan, sehingga untuk beberapa kasus, metode PSO klasik menjadi kurang efisien. Perbaikan PSO dengan menerapkan *Linear Decreasing Inertia Weight* (LDIW) yang berfungsi mengontrol kecepatan dari waktu ke waktu sehingga kecepatan mengalami penurunan secara linier yang membuat PSO mencari ruang yang lebih besar di awal untuk mendapatkan posisi dengan cepat yaitu solusi yang paling optimis. Saat *inertia weight* menurun, kecepatan partikel juga ikut diperlambat untuk mencari solusi parsial yang lebih halus sehingga metode perbaikan ini mampu memberikan keseimbangan antara eksplorasi global dan eksploitasi lokal [11].

Perbaikan dan modifikasi lainnya pada PSO adalah menggunakan *Constriction Factor Method* (CFM) untuk menjamin konvergensi dan osilasi amplitudo partikel menurun dari waktu ke waktu tanpa pengaturan kecepatan maksimum [12]. Atas dasar ide tersebut peneliti menerapkan kedua metode tersebut untuk mencari solusi penentuan disain fungsi keanggotaan fuzzy yang optimal.

Penelitian ini melakukan pendekatan heuristik berbasis algoritma MPSO serta meneliti perbedaan metode LDIW dan CFM dalam mengoptimalkan fungsi keanggotaan *fuzzy*.

2. MODIFIED PARTICLE SWARM OPTIMIZATION (MPSO)

Persamaan PSO klasik diubah untuk meningkatkan kemampuan optimasi PSO. Kelompok pertama dari modifikasi PSO terdiri modifikasi terhadap parameter bobot inersia atau *Linear Decreasing Inertia Weight* (LDIW) dan yang kedua menggunakan parameter faktor penyempitan atau *Constriction Factor Method* (CFM) [13].

2.1. Linear Decreasing Inertia Weight (LDIW)

Pada metode PSO klasik, nilai bobot inersia atau *inertia weight* dibuat konstan, sehingga untuk beberapa kasus, metode PSO klasik menjadi kurang efisien. Shi dan Eberhart (1998) melakukan modifikasi terhadap nilai bobot inersia dengan pertimbangan pada saat awal iterasi, bobot inersia diatur dengan nilai yang cukup besar untuk memperluas daerah pencarian dan menghindari terjebak di *local optimum*, kemudian pada iterasi terakhir, bobot inersia diatur cukup kecil untuk mendapatkan hasil akhir yang akurat. menyatakan bobot inersia dengan metode LDIW dapat ditentukan menggunakan persamaan (1) [14].

$$w_k^i = \frac{(w_{max} - w_{min})}{iter_{max}} * iter \quad (1)$$

Selanjutnya [14] mendeskripsikan penentuan kecepatan *particle* pada MPSO dapat ditentukan menggunakan persamaan (2.13) berikut ini :

$$v_{k+1}^i = w_k^i * v_k^i + c_1 * rand * (p^i - x_k^i) + c_2 * rand * (p_g^i - x_k^i) \quad (2)$$

dimana :

w_k^i = inertia weight pada epoch ke- i

w_{max} = inertia weight maksimum

w_{min} = inertia weight minimum

$iter_{max}$ = epoch maksimum

$iter$ = current epoch

Parameter w_{max} biasanya diterapkan sebesar 0,9 dan parameter w_{min} diterapkan sebesar 0,4. Parameter $C_1 = C_2 = 2$ [9].

2.2. Constriction Factor Method (CFM)

Clerc dalam [14] melakukan penerapan *constriction factor* yang dikenal dengan istilah *Constriction Factor Method* (CFM). Peningkatan dan modifikasi ini bertujuan untuk menjamin suatu penelusuran dalam algoritma PSO untuk konvergen lebih cepat. *Constriction factor* ditentukan dengan menggunakan persamaan (3).

$$C = \frac{2}{|2 - \phi - \sqrt{\phi^2 - 4\phi}|} \quad (3)$$

Kecepatan *particle* pada MPSO dapat ditentukan menggunakan persamaan (4) berikut ini :

$$v_{k+1}^i = C * \{v_k^i + c_1 * rand * (p^i - x_k^i) + c_2 * rand * (p_g^i - x_k^i)\} \quad (4)$$

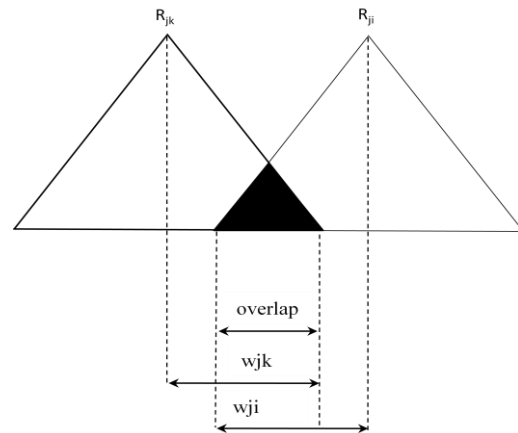
Parameter $\phi = C_1 + C_2$ dan $\phi > 4$, sehingga untuk memenuhi syarat tersebut nilai C_1 dan C_2 biasanya bernilai 2, 05.

3. FUNGSI EVALUASI (FITNESS FUNCTION)

Nilai *fitness* pada sebuah fungsi keanggotaan *fuzzy* dapat dihitung dengan mempertimbangkan keterhubungan daerah linguistik yang satu dengan daerah linguistik yang lainnya [15]. Terdapat dua kriteria evaluasi yang dapat digunakan untuk mengevaluasi disain fungsi keanggotaan *fuzzy* yaitu faktor *overlap* dan faktor *coverage*.

3.1. Faktor Overlap

Perbandingan *overlap* dari fungsi keanggotaan R_{jk} dan R_{ji} dapat didefinisikan sebagai panjang *overlap* dibagi setengah jarak minimum dari dua fungsi tersebut. Pada gambar 1 ditampilkan gambaran keadaan *overlap* dua daerah *fuzzy* segitiga.

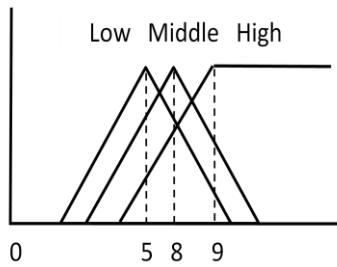


Gambar 1. *Overlap* dua daerah segi tiga

Faktor *overlap* pada dua daerah *fuzzy* dapat ditentukan menggunakan persamaan (5) berikut ini :

$$= \sum_{k \neq i} \left[\max \left(\left(\frac{overlap(R_{jk}, R_{ji})}{\min(w_{jk}, w_{ji})} \right), 1 \right) - 1 \right] \quad (5)$$

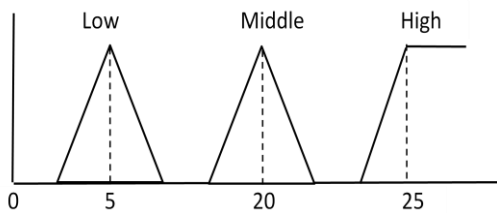
Pada persamaan (5) di atas, *overlap* (R_{jk}, R_{ji}) adalah jarak *overlap* antara daerah R_{jk} dan R_{ji} . Faktor *overlap* didesain untuk menghindari supaya fungsi keanggotaan *fuzzy* tidak tumpang tindih terlalu dekat seperti terlihat pada gambar 2.



Gambar 2. Bentuk fungsi keanggotaan saling overlap terlalu dekat

3.3. Faktor Coverage

Faktor *coverage* didesain untuk menghindari supaya fungsi keanggotaan tidak terpisah terlalu jauh seperti gambar 3 berikut ini :



Gambar 3. Bentuk fungsi keanggotaan yang terpisah terlalu jauh

Faktor *coverage* fungsi keanggotaan I_j dapat didefinisikan pada persamaan (6) berikut ini :

$$\text{Faktor coverage} = \frac{1}{\frac{\text{range}(R_{jk}, R_{ji})}{\max(I_j)}} \quad (6)$$

3.4. Faktor Suitability

Faktor Suitabilitas (*suitability*) adalah hasil penjumlahan dari faktor *overlap* dengan faktor *coverage* sebagaimana dirumuskan pada persamaan (7) berikut ini :

$$\sum_{k \neq i} \left[\max \left(\left(\frac{\text{overlap}(R_{jk}, R_{ji})}{\min(w_{jk}, w_{ji})} \right), 1 \right) - 1 \right] + \frac{1}{\frac{\text{range}(R_{jk}, R_{ji})}{\max(I_j)}} \quad (7)$$

Untuk menentukan nilai *fitness* dapat ditentukan dengan persamaan (8) berikut ini :

$$\text{Fitness} = \frac{1}{\text{suitability}} \quad (8)$$

4. VARIABEL DAN DATASET.

Fungsi keanggotaan *fuzzy* terdiri dari 15 parameter dan 5 daerah linguistik yaitu Sangat Rendah (SR), Rendah (R), Cukup (C), Tinggi (T) dan Sangat Tinggi (ST). *Dataset* yang digunakan dalam penelitian ini terdiri dari dua variabel yaitu :

1. Variabel Produktifitas : range 20 - 70
2. Variabel Keterisolasian : range 12 - 80

5. PARAMETER MPSTO

5.1. Parameter LDIW

Parameter yang digunakan pada metode ini adalah *inertia weight* awal (W_{up}), *inertia weight* akhir (W_{low}), faktor *learning particle* (C_1), faktor *learning swarm* (C_2), jumlah *particle* jumlah *swarm*. Parameter-parameter tersebut ditampilkan pada tabel 1 dan tabel 2. Tabel 1 digunakan untuk pengujian 50 partikel sedangkan tabel 2 digunakan untuk pengujian 20 partikel

Tabel 1. Parameter LDIW 50 Particle

W_{Up}	W_{Low}	C_1	C_2	<i>Swarm</i>	<i>Particle</i>
0,9	0,4	2	2	100	50

Tabel 2. Parameter LDIW 20 Particle

W_{Up}	W_{Low}	C_1	C_2	<i>Swarm</i>	<i>Particle</i>
0,9	0,4	2	2	100	20

5.2. Parameter CFM

Parameter yang digunakan pada metode ini adalah faktor *learning particle* (C_1), faktor *learning swarm* (C_2), jumlah *particle* serta jumlah *swarm*. Parameter-parameter tersebut ditampilkan pada tabel 3 dan 4. Tabel 3 digunakan untuk pengujian 50 partikel sedangkan tabel 4 digunakan untuk pengujian 20 partikel

Tabel 3. Parameter CFM 50 Particle

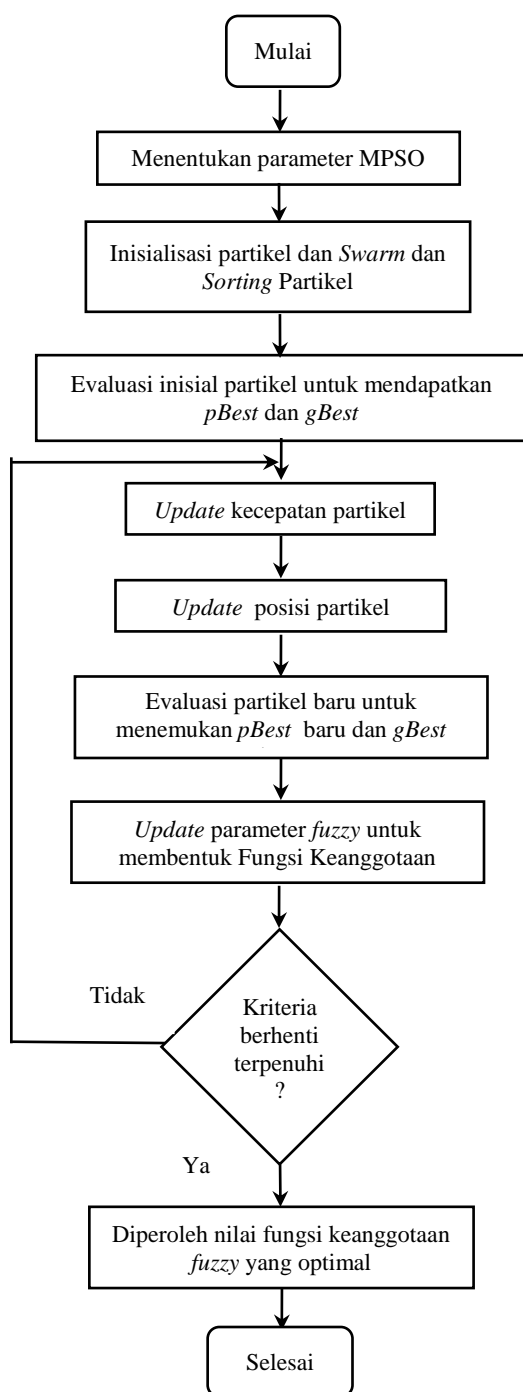
C_1	C_2	<i>Swarm</i>	<i>Particle</i>
2, 05	2, 05	100	50

Tabel 4. Parameter CFM 20 Particle

C_1	C_2	<i>Swarm</i>	<i>Particle</i>
2, 05	2, 05	100	20

6. PERANCANGAN ALGORITMA

Disain diagram alir optimasi fungsi keanggotaan *fuzzy* menggunakan algoritma MPSTO dapat digambarkan sebagaimana terlihat pada gambar 4.

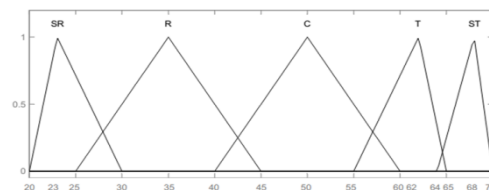


Gambar 4. Algoritma Optimasi Fuzzy Menggunakan Algoritma MPSO

7. REPRESENTASI PARTIKEL

7.1. Representasi Partikel Produktifitas

Variabel Produktifitas dengan *range* 20-70 dan lima daerah *fuzzy* yaitu SR(20, 23, 30), R(25, 35, 45), C(40, 50, 60), T(55, 62, 65) dan ST(64, 68, 70).



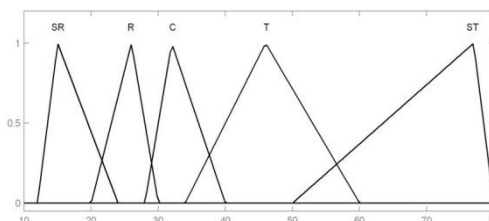
Gambar 5. Representasi partikel fungsi keanggotaan produktifitas

Tabel 5. Representasi partikel produktifitas

Partikel
20,23,30,25,35,45,40,50,60,55,62,65,64,68,70

7.2. Representasi Partikel Keterisolasian

Variabel Keterisolasian dengan *range* 20-70 dan lima daerah *fuzzy* yaitu SR(12, 15, 24), R(20, 26, 30), C(28, 32, 40), T(34, 46, 60) dan ST(50, 77, 80).



Gambar 6. Representasi partikel fungsi keanggotaan keterisolasian

Grafik fungsi keanggotaan pada gambar 6 di atas dapat direpresentasikan ke dalam bentuk partikel seperti pada tabel 6 berikut :

Tabel 6. Representasi partikel keterisolasian

Partikel
12,15,24,20, 26,30,28,32,40,34,46,60,50,77, 80

8. HASIL PENGUJIAN

8.1. Inisialisasi Swarm

Tabel 7 dan 8 berikut ini adalah gambaran (sebahagian) inisial *swarm* atau *swarm* pertama variabel Produktifitas dan variabel Keterisolasian. Variabel Produktifitas diinisialisasi dengan *range* 0-70 dan variabel Keterisolasian diinisialisasi dengan *range* 0-80. *Swarm* pertama terdiri atas 50 *particle* (P1-P50) dan masing-masing *particle* memiliki nilai *fitness* yang berbeda.

Tabel 7. Inisialisasi *Swarm* Produktifitas

P	Particle															Fitness
P1	1	4	13	9	17	31	23	35	40	40	42	49	48	56	63	6,45
P2	3	7	18	12	26	40	39	42	52	49	53	63	55	63	68	5,45
P3	0	1	10	2	12	25	15	25	36	26	38	57	41	59	64	7,48
P4	9	21	36	34	39	54	40	58	60	60	61	62	62	67	67	4,41
P5	2	6	14	13	17	28	24	29	29	29	35	51	44	53	64	7,42
P10	0	0	2	1	15	19	16	23	26	25	29	42	30	45	49	6,46
P48	3	9	25	14	25	35	34	40	51	42	54	61	59	63	67	5,45
P49	3	4	13	13	16	30	22	43	54	51	56	57	56	58	60	6,47
P50	2	9	16	16	16	26	24	30	38	36	50	64	62	64	66	6,45

Tabel 8. Inisialisasi *Swarm* Keterisolasian

P	Particle															Fitness
P1	15	15	28	19	35	49	47	50	57	53	57	71	63	72	75	4,43
P2	2	20	23	22	29	41	32	42	49	44	53	68	67	70	74	3,40
P3	3	9	14	11	14	19	18	21	41	23	65	72	71	77	79	5,46
P4	1	3	5	3	12	36	27	42	55	44	61	64	64	76	78	5,48
P5	0	7	32	17	37	39	38	44	54	44	58	63	60	63	73	4,43
P10	2	13	24	16	30	35	34	39	43	42	58	76	70	79	79	5,46
P48	1	20	25	24	34	44	38	46	47	46	53	70	59	70	73	5,48
P49	4	8	14	9	42	51	50	54	64	58	66	69	66	70	75	5,43
P50	17	21	24	22	32	39	33	51	53	53	64	71	68	72	77	6,46

8.2. Local Best

Local best fitness adalah nilai *fitness* terkecil pada setiap *swarm*. Pada tabel berikut ini ditampilkan (sebahagian) *local best fitness* untuk *swarm* pertama Produktifitas.

Tabel 9. *Local Best Fitness Swarm* Pertama

P	Fitness	P	Fitness
P1	6,45	P26	3,37
P2	5,45	P27	5,4
P11	6,48	P36	4,47
P12	6,44	P37	6,47
P13	6,46	P38	5,45
P14	7,47	P39	7,43
P15	7,46	P40	6,48
P16	4,44	P41	4,46
P17	5,47	P42	3,4
P18	6,47	P43	7,46
P23	5,47	P48	5,45
P24	6,44	P49	6,47
P25	5,43	P50	6,45
Min			3,37

Pada tabel 9 di atas, *fitness* terbaik (terkecil) untuk *swarm* pertama adalah 3,37 yaitu pada *swarm* 26 (P26).

8.3. Global Best

Global best adalah *particle* paling baik dengan nilai *fitness* paling kecil di antara semua *swarm* yang ada. Untuk variabel produktifitas, *Global best* terdapat pada *swarm* 91, *particle* 41 dan dengan nilai *fitness* sebesar 1,4. Tabel 10 berikut ini adalah partikel terbaik variabel Produktifitas dengan range 20-70. Angka-angka tersebut akan dipakai sebagai paramater pembentuk grafik fungsi keanggotaan fuzzy teroptimasi.

Tabel 10. *Global Best Fitness* Produktifitas

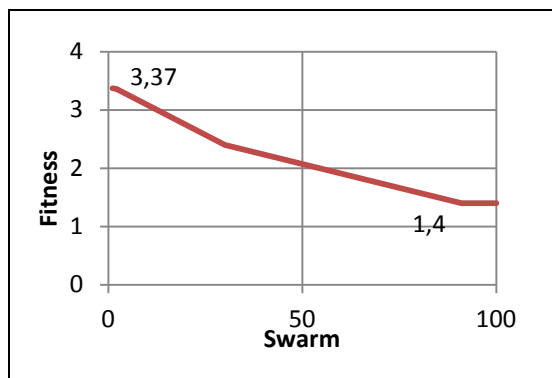
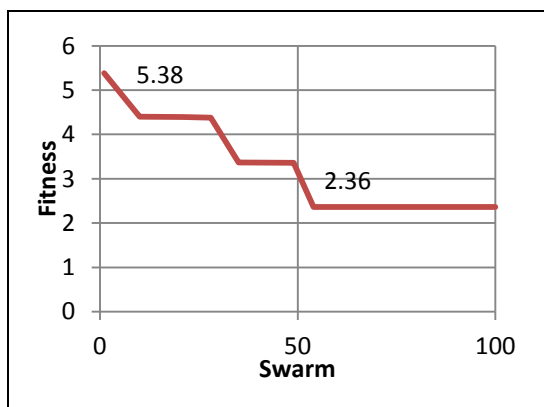
Particle
P41 20,22,28,25,37,47,37,48,53,52,61,63,62,69,70

Tabel 11. *Global Best Fitness* Keterisolasian

Particle
P7 12,17,25,17,28,46,39,47,48,47,62,70,64,70,80

8.4. Global Seeking

Global seeking adalah proses penelusuran *global best fitness* dimulai dari *swarm* pertama. *Global best fitness* merupakan *fitness* dengan nilai paling kecil, sehingga proses penelusuran akan membentuk grafik menurun sebagaimana ditampilkan pada gambar 7.

Gambar 7. Grafik *Global Seeking* ProduktifitasGambar 8. Grafik *Global Seeking* Keterisolasian

Pada gambar 7 di atas, variabel Produktifitas memiliki *local best fitness* sebesar 3,37 pada *swarm* awal dan menemukan *global best fitness* sebesar 1,40 pada *swarm* 91(S91). Pada gambar 8 di atas, variabel Keterisolasian memiliki *local best fitness* sebesar 3,39 pada *swarm* awal dan menemukan *global best fitness* sebesar 1,40 pada *swarm* 84 (S84).

9. PERBANDINGAN HASIL OPTIMASI

Pada tabel 7 dan 8 ditampilkan hasil keseluruhan pencapaian *global best fitness* masing-masing variabel. Pada tabel tersebut ditampilkan keseluruhan nilai *global best fitness* yang diperoleh padametode LDIW dan CFM. Keseluruhan nilai *global best fitness* lebih kecil atau lebih baik dari nilai *fitness* sebelum dioptimasi

Tabel 12. Perbandingan optimasi produktifitas

Perco-baan	Metode			
	LDIW 50	LDIW 20	CFM 50	CFM 20
1	1,4	2,34	1,4	2,34
2	1,4	2,37	1,4	2,37
3	1,4	2,34	1,4	2,34
4	1,4	2,34	1,4	2,34
5	1,4	2,34	1,4	2,34
6	1,4	2,34	1,4	2,34
7	1,4	2,34	1,4	2,34
8	1,4	2,37	1,4	2,37
9	1,4	2,34	1,4	2,34
10	1,4	2,34	1,4	2,37
Sebelum	3,03			

Tabel 13. Perbandingan optimasi keterisolasian

Perco-baan	Metode			
	LDIW 50	LDIW 20	CFM 50	CFM 20
1	1,4	3,37	1,4	2,4
2	1,4	3,39	1,4	2,4
3	1,4	3,38	1,4	2,4
4	1,4	3,38	1,4	2,4
5	1,4	3,37	2,35	2,4
6	1,4	3,38	1,4	3,4
7	1,4	3,38	2,37	2,4
8	1,4	3,38	1,4	2,4
9	1,4	3,38	1,4	2,4
10	1,4	3,39	1,4	2,4
Sebelum	4,16			

Pada tabel 12 di atas diperoleh *global best fitness* hasil optimasi dengan *range* 1,4 -2,37, nilai *fitness* sebelum dioptimasi adalah 3, 03. Hal tersebut menunjukkan bahwa dengan metode MPSO diperoleh optimasi atau perbaikan nilai *fitness* fungsi keanggotaan fuzzy Produktifitas. Pada tabel 8 di atas diperoleh *global best fitness* hasil optimasi dengan *range* 1,4 - 3,39, nilai *fitness* sebelum dioptimasi adalah 4, 16. Hal tersebut menunjukkan bahwa dengan metode MPSO diperoleh optimasi atau perbaikan nilai *fitness* fungsi keanggotaan fuzzy Keterisolasian.

10. PERBANDINGAN METODE OPTIMASI

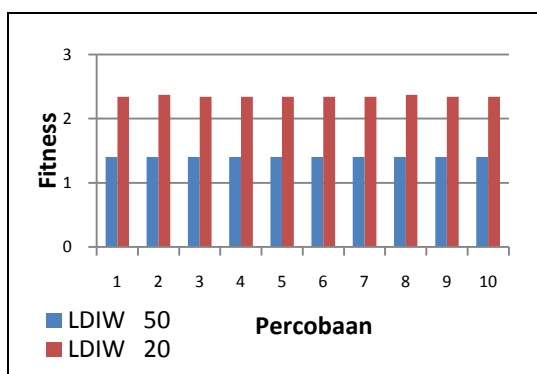
Pada penelitian ini penentuan kecepatan *particle* dilakukan dengan dua cara yaitu dengan metode LDIW dan dengan CFM. Masing-masing metode tersebut diuji sebanyak 10 kali pada dua jenis *particle* yang berbeda yakni 50 *particle* dan 20 *particle*.

9.1. LDIW 50 particle dengan LDIW 20 Particle

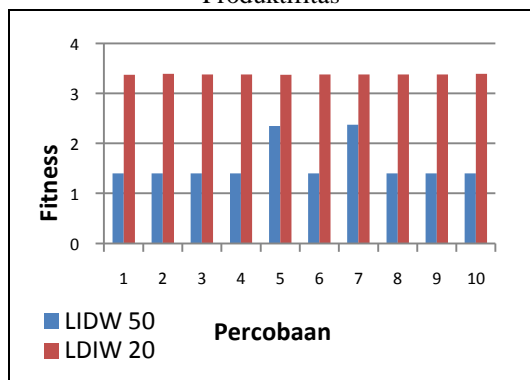
Tabel 14. Perbandingan LDIW 50 Particle dan LDIW 20 Particle

Perco- baan	Produktifitas		Keterisolasian	
	LDIW 50	LDIW 20	LDIW 50	LDIW 20
1	1,4	2,34	1,4	3,37
2	1,4	2,37	1,4	3,39
3	1,4	2,34	1,4	3,38
4	1,4	2,34	1,4	3,38
5	1,4	2,34	2,35	3,37
6	1,4	2,34	1,4	3,38
7	1,4	2,34	2,37	3,38
8	1,4	2,37	1,4	3,38
9	1,4	2,34	1,4	3,38
10	1,4	2,34	1,4	3,39

Tabel 14 di atas ditampilkan hasil perbandingan *global best fitness* MPSO metode LDIW 50 particle dengan LDIW 20 particle pada pengujian sebanyak 10 kali percobaan. Tabel tersebut menunjukkan bahwa dengan metode LDIW 50 particle diperoleh nilai *fitness* yang lebih minimum jika dibandingkan dengan metode LDIW 20 particle. Hasil pada tabel 9 di atas dapat ditampilkan dalam bentuk grafik pada gambar 9 dan gambar 10 berikut ini :



Gambar 9. Grafik Perbandingan *Global Best Fitness* Produktifitas



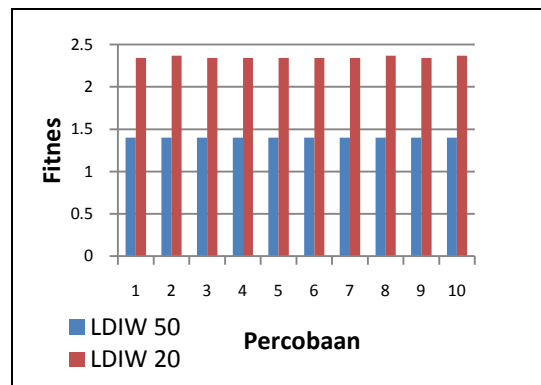
Gambar 10. Grafik Perbandingan *Global Best Fitness* Keterisolasian

9.2. CFM 50 Particle dengan CFM 20 Particle

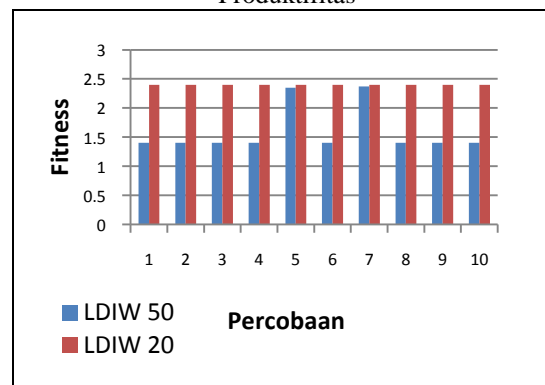
Tabel 15. Perbandingan CFM 50 Particle dengan CFM 20 Particle

Perco- baan	Produktifitas		Keterisolasian	
	LDIW 50	LDIW 20	LDIW 50	LDIW 20
1	1,4	2,34	1,4	2,4
2	1,4	2,37	1,4	2,4
3	1,4	2,34	1,4	2,4
4	1,4	2,34	1,4	2,4
5	1,4	2,34	2,35	2,4
6	1,4	2,34	1,4	2,4
7	1,4	2,34	2,37	2,4
8	1,4	2,37	1,4	2,4
9	1,4	2,34	1,4	2,4
10	1,4	2,37	1,4	2,4

Tabel 10 menampilkan hasil perbandingan optimasi MPSO metode CFM 50 particle dengan CFM 20 particle pada pengujian sebanyak 10 kali percobaan. Tabel tersebut menunjukkan bahwa dengan metode CFM 50 particle didapat nilai *fitness* yang lebih optimal dibandingkan dengan metode CFM 20 particle. Hasil pada tabel 10 di atas dapat ditampilkan dalam bentuk grafik pada gambar 11 dan gambar 12 berikut ini :



Gambar 11. Grafik Perbandingan *Global Best Fitness* Produktifitas



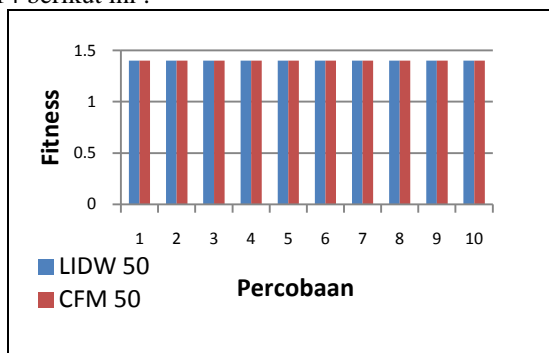
Gambar 12. Grafik Perbandingan *Global Best Fitness* Keterisolasian

10.1. LDIW 50 Particle dengan CFM 50 Particle

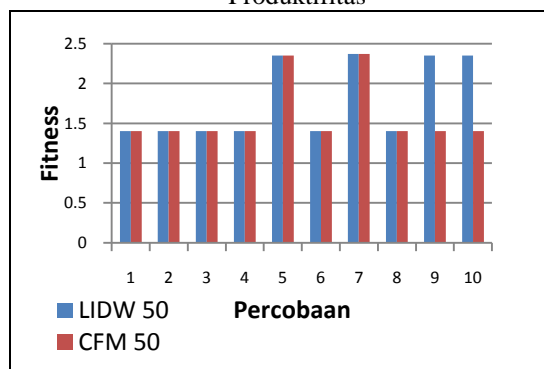
Tabel 16 Perbandingan LDIW 50 Particle dengan CFM 50 Particle

Percobaan	Produktifitas		Keterisolasian	
	LDIW 50	CFM 50	LDIW 50	CFM 50
1	1,4	1,4	1,4	1,4
2	1,4	1,4	1,4	1,4
3	1,4	1,4	1,4	1,4
4	1,4	1,4	1,4	1,4
5	1,4	1,4	2,35	2,35
6	1,4	1,4	1,4	1,4
7	1,4	1,4	2,37	2,37
8	1,4	1,4	1,4	1,4
9	1,4	1,4	2,35	1,4
10	1,4	1,4	2,35	1,4

Tabel 16 menampilkan hasil perbandingan *global best fitness* MPSO metode LDIW 50 particle dengan CFM 50 particle. Tabel tersebut menunjukkan bahwa metode CFM didapat nilai *fitness* yang lebih kecil atau sama dibanding dengan metode LDIW, dengan kata lain hasil pencapaian *global best fitness* pada metode LDIW tidak pernah melampaui hasil pencapaian *global best fitness* pada metode CFM. Hasil pada tabel 11 di atas dapat ditampilkan dalam bentuk grafik pada gambar 13 dan 14 berikut ini :



Gambar 13. Grafik Perbandingan *Global Best Fitness* Produktifitas



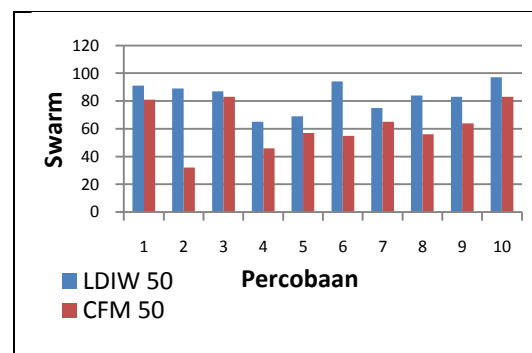
Gambar 14. Grafik Perbandingan *Global Best Fitness* Keterisolasian

10.2. Perbandingan Swarm LDIW 50 Particle dengan CFM 50 Particle

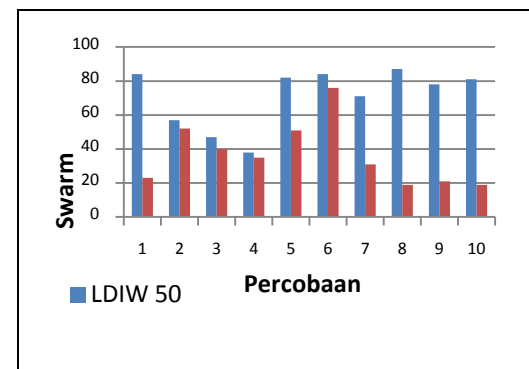
Selain mengamati dari sisi perbandingan *global best fitness*, hal lain yang juga diamati pada perbandingan metode LDIW dan CFM adalah perbandingan *swarm*. Tabel 12 menampilkan hasil perbandingan *swarm* metode LDIW 50 particle dengan CFM 50 particle. Tabel tersebut menunjukkan bahwa dengan metode CFM 50 particle ditemukan solusi yang lebih cepat atau *swarm* yang lebih sedikit jika dibandingkan dengan metode LDIW 50 particle

Tabel 17 Perbandingan Swarm LDIW 50 Particle dengan CFM 50 Particle

Percobaan	Produktifitas		Keterisolasian	
	LDIW 50	CFM 50	LDIW 50	CFM 50
1	91	81	84	23
2	89	32	57	52
3	87	83	47	40
4	65	46	38	35
5	69	57	82	51
6	94	55	84	76
7	75	65	71	31
8	84	56	87	19
9	83	64	78	21
10	97	83	81	19



Gambar 15. Grafik Perbandingan Swarm

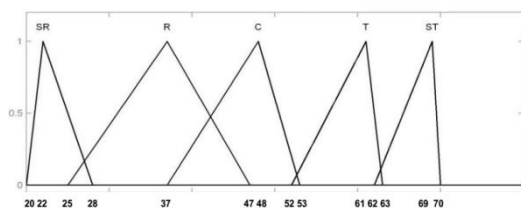


Gambar 16. Grafik Perbandingan Swarm Metode LDIW 50 Particle dengan CFM 50 Particle

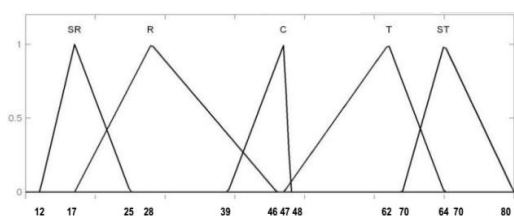
11. GRAFIK FUNGSI KEANGGOTAAN TEROPTIMASI

Penelitian ini bertujuan untuk menentukan parameter dan disain fungsi keanggotaan *fuzzy* yang optimal. Gambar 17, 18, 19 dan 20 adalah disain fungsi keanggotaan yang optimal untuk variabel produktifitas dan keterisolasian

10.1 LDIW 50 Particle



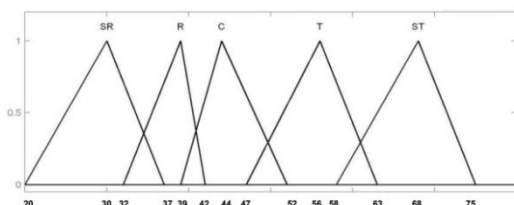
(a)



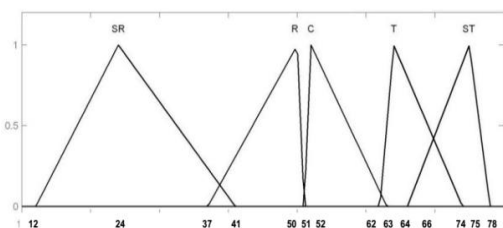
(b)

Gambar 17. Grafik fungsi keanggotaan teroptimasi (a) Produktifitas (b) Keterisolasian

10.2 LDIW 20 Particle



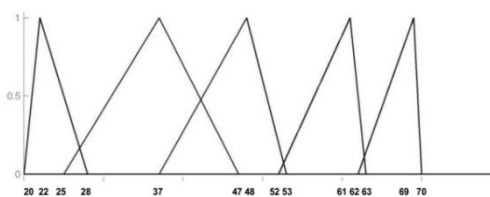
(a)



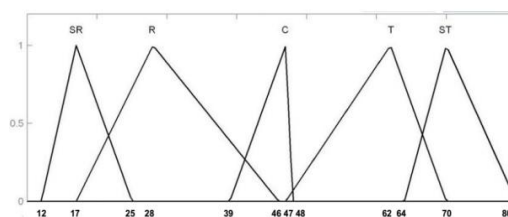
(b)

Gambar 18. Grafik fungsi keanggotaan teroptimasi (a) Produktifitas (b) Keterisolasian

10.3 CFM 50 Particle



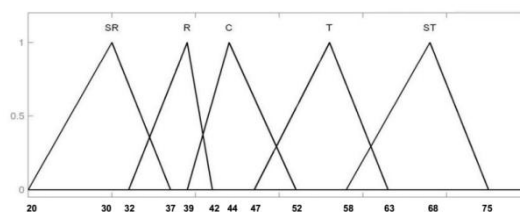
(a)



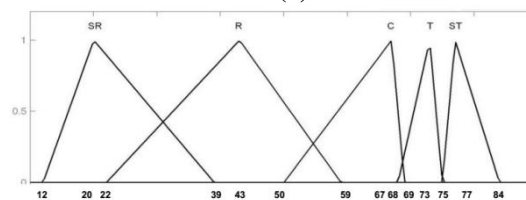
(b)

Gambar 19. Grafik fungsi keanggotaan teroptimasi (a) Produktifitas (b) Keterisolasian

10.3 CFM 20 Particle



(a)



(b)

Gambar 20. Grafik fungsi keanggotaan teroptimasi (a) Produktifitas (b) Keterisolasian

12. PENUTUP

Berdasarkan percobaan yang telah dilakukan untuk mengoptimasi fungsi keanggotaan *fuzzy* menggunakan algoritma MPSO dengan metode *Linear Decreasing Inertia Weight* (LDIW) dan *Constriction Factor Method* (CFM) didapat kesimpulan bahwa Dalam hal ini semakin banyak jumlah *particle* maka didapatkan ruang pencarian solusi yang lebih luas sehingga diperoleh nilai *fitness* yang lebih optimal. Untuk jumlah *particle* yang sama, hasil *global best fitness* yang diperoleh dengan metode CFM lebih optimal dibandingkan dengan metode LDIW. Secara keseluruhan, nilai *global best fitness* pada metode LDIW tidak pernah melebihi keoptimalan CFM. Untuk jumlah *particle* yang sama,

metode CFM mencapai solusi yang lebih cepat dibandingkan dengan LDIW, dengan kata lain *particle* pada FCM mencapai konvergensi yang lebih cepat dibandingkan dengan metode LDIW.

Pada penelitian lain yang berkaitan dengan optimasi fungsi keanggotaan *fuzzy* menggunakan algoritma MPSO, pengujian perlu dilakukan dengan melibatkan jumlah *particle* dan *swarm* yang lebih banyak serta parameter yang lebih *variatif* dari yang sudah diterapkan pada penelitian ini.

DAFTAR PUSTAKA

- [1] F.B. Alwi, Knowledge acquisition tool for learning membership function and fuzzy classification rules from numerical data. *International Journal of Computer Applications (IJCA)* vol. 64, no13, pp. 24-30, 2013.
- [2] S.Wang, G. Wang, M. Gao, and S. Yu, Using fuzzy hybrid features to classify strokes in interactive sketches. *Journal Of Advances in Mechanical Engineering*, vol. 25, no. 3, pp. 1-7, 2013.
- [3] M.N. Vikas and R.K. Prabhas, 2. Efficient hand off using fuzzy and simulated annealing. *Internatioanl Journal of Computer Network and Information Security* , vol. 12, no. 1, pp. 17-23, 2012
- [4] A.M. Acilar and A. Arslan, Optimization of multiple input single output fuzzy membership functions using clonal selection algorithm. *Proceedings of the 8th WSEAS International Conference on Applied Computer Science (ACS'08)*, pp. 49-53, 2008
- [5] S.J. Sruthi, Self generated fuzzy membership function using ANN clustering technique. *International Journal of Latest Trends in Engineering and Technology* , vol. 13, no. 2, pp. 142-152, 2013.
- [6] A .Khosla, Comparison of ABC and ant colony algorithm based fuzzy controller for an inverted pendulum. *International Journal of Innovative Technology and Exploring Engineering*, vol. 3, no. 3 , pp. 123-234, 2013.
- [7] D. Dutta, R. Roy and K. Choudhury, Training artificial neural network using particle swarm. *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no 3, pp. 430- 434, 2013.
- [8] M. Nayak, GPAC-APSO Clustering using modified s-transform for data mining. *International Journal Of Engineering Science & Advanced Technology*, vol. 2, no.1, pp. 38 – 48, 2012.
- [9] C.H. Yang, C-H. Hsiao and L-Y Chuang, Linearly Decreasing Weight Particle Swarm Optimization with AcceleratedStrategy for ata Clustering . *International Journal of Computer Science*, vol. 37, no. 3, pp. 3-9, 2010.
- [10] Lane, M.C., Xue, B. & Liu, I. 2013. PSO based selection of spectral features for remotely sensed image classification. *International Journal on Artificial Intelligence Tools* 22 (4) : 214-218.
- [11] Y. Shi and R.C. Eberhart, A modified particle swarm optimizer. *Proceedings of Congress on Evolu-tionary Computation*, pp. 79-73, 1998
- [12] M. Clerc, The Swarm and The Queen: Towards a Deterministic and Adaptive Particle Swarm Optimization, *Proceeding of Congress on Evolutionary Computation, ashington*, pp. 1951-1957, 1999
- [13] M. Jacobcoca, P. Maca and P. Pech, A Comparison of Selected Modifications of the Particle Swarm Optimization Algorithm. *Journal of Applied Mathematics*, vol. 14, no.2014, pp. 10-15, 2014
- [14] H. Guo and J. He, A modified particle swarm optimization algorithm. *Journal of Computer Science* , vol. 10, no 2, pp. 341-346, 2013.
- [15] T.P. Hong, C.H. Chen, Y.C. Lee and Y.L Wu, Genetic-fuzzy data mining with divide and conquer strategi. *IEEE Transaction On Evolutionary Computation*, vol. 12 , no. 2, pp. 252-265, 2008.